

REMARKS

This amendment is submitted in response to the Office Action dated July 28, 2005. Claims 1 and 9-17 are currently amended. Applicant has amended the claims to clarify key features of the invention and overcome the claim objections and rejections. The amendments place the claims in better condition for allowance. Applicants respectfully requests entry of the amendments to the claims. The discussion and arguments provided below reference the claims in their amended form.

CLAIM OBJECTIONS

In the present Office Action, Claims 1, 9, and 17 are objected to because of insufficient antecedent basis with respect to the phrase 'the set consisting of'. Applicant has amended the claims to remedy this informality. Applicant appreciates the Examiner's attention to detail.

CLAIMS REJECTIONS UNDER 35 U.S.C. § 101

In the present Office Action, Claims 9-16 are rejected under 35 U.S.C. § 101 due to the recitation of 'a computer program product'. Applicant has amended the claims to more clearly recite allowable statutory subject matter.

II. Introduction to Rejections under 35 U.S.C. § 103

In the present Office Action, Claims 1-3, 7-11, 15-19, and 23-24 are rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,999,971 to Buckland (*Buckland*) in view of U.S. Patent No. 5,774,660 to Brendel *et al.* (*Brendel*). Claims 10-11, 15-16, 18-19 and 23-24 have similar limitations as Claims 2-3 and 7-8; therefore, they are rejected under the same rationale. Claims 4-6, 12-14, and 20-22 are rejected under 35 U.S.C. § 103(a) as unpatentable over *Buckland* and *Brendel* and in view of U.S. Patent No. 5,813,007 to Nielsen *et al.* (*Nielsen*). Those rejections are respectfully traversed in view of the discussion made herein, and favorable reconsideration of the claims is requested.

III. The ‘determining’ step of exemplary Claim 1 is not taught or suggested

First, with respect to exemplary Claim 1, Applicant respectfully submits that the cited combination of references does not teach or suggest Applicant’s claimed feature of “determining, based on the recency of a time stamp contained within a client’s request to receive a file from a content server, whether said client's request to receive said file from said content server originated as a reference from one of a set consisting of the load distribution server and the content server”. Use of the recency of a time stamp contained within a client’s request to receive a file from a content server is supported, *inter alia* at page 15 and line 1 of the original specification. The Examiner asserts at paragraph 14 that **Buckland** “teaches determining whether a client's request to receive a file from a content server originated as a reference from the content server itself”. The Examiner cites **Figure 3** of *Buckland*, as well as Column 6, lines 5-15 as teaching this functionality. The cited text of *Buckland* discloses: :

The process begins at step 300 in which the client 206 requests access to the first network site 200. Although the first network site 200 is discussed, the principles relating to this process can be applied to such a request received by any of the other network sites. Once the request is received by the first network site 200, then the process continues to step 302 in which the first network site 200 interacts with the client 206, in a conventional manner, to determine if the browser 210 includes a first site cookie (i.e., first network site data block) from the first network site 200. As is known in the art, the browser 210 may include a first site cookie if that browser 210 had accessed the first site at some earlier time and the first site transmitted (a/k/a "dropped") a cookie to the browser 210 for subsequent retrieval by the first network site 200. *Buckland* (Column 6, lines 1-15)

Having reviewed the cited references and amended exemplary Claim 1, Applicant respectfully submits that the cited texts do not teach or suggest “determining, based on the recency of a time stamp contained within a client’s request to receive a file from a content server, whether said client's request to receive said file from said content server originated as a reference from one of a set consisting of the load distribution server and the content server”, as is recited in amended exemplary Claim 1. More specifically, the use of the recency of a time stamp contained within a client’s request to receive a file from a content server is not taught or suggested in *Buckland*, or in any of the cited references, and none of the uses of the word ‘time’ in *Buckland* could be reasonably construed to suggest such a teaching. Further, the combination

of the references does not teach or suggest the use of the recency of a time stamp contained within a client's request to receive a file from a content server as recited in Applicants' amended exemplary claim 1.

Applicant respectfully submits that no reasonable combination of the cited references teaches or suggests Applicant's claimed feature of "determining, based on the recency of a time stamp contained within a client's request to receive a file from a content server, whether said client's request to receive said file from said content server originated as a reference from one of a set consisting of the load distribution server and the content server". Applicant respectfully submits that the feature recited in the cited art, of seeking a cookie from a client under the teaching of *Buckland* does not teach or suggest "determining, based on the recency of a time stamp contained within a client's request to receive a file from a content server, whether said client's request to receive said file from said content server originated as a reference from one of a set consisting of the load distribution server and the content server". Applicant most respectfully asserts that ascertaining the presence or absence of the cookie in *Buckland* is inadequate to determine, and does not suggest determining, the referring source of the request, based on the recency of a time stamp.

Applicant respectfully notes that, while the cookie of *Buckland* allows a server to know if it has contacted a client before, the cited references neither teach nor suggest Applicant's claimed feature of "determining whether a client's request to receive a file from a content server originated as a reference from one of the set consisting of the load distribution server and the content server," as is recited in exemplary Claim 1. In view of the absence, when considering the combination of all of the cited references, of the recited functionality, Applicant respectfully submits that the rejection under 35 U.S.C. § 103 is overcome.

IV. The 'sending' step of exemplary Claim 1 is not taught or suggested

Still with respect to Claim 1, Applicant most respectfully submits that the cited combination of references does not teach or suggest Applicant's claimed feature of "responsive to determining that the client's request to receive the file from the content server did not originate as the reference from one of the set consisting of the load distribution server and the content server, sending to the client a file requesting that the client contact the load distribution server",

as recited in Applicant's exemplary Claim 1. While the Examiner concedes that "Buckland does not specially disclose a load distribution server", the Examiner cites *Buckland*, at Column 6, lines 25-50 as teaching related functionality. The Examiner also cites *Brendel* as teaching a load balancer functionality, as is discussed below. The cited portion of *Buckland* discloses:

If, however, it was determined at step 302 that the browser 210 did not include a first site cookie, then the process continues to step 306 in which the browser 210 is redirected (also referred to as "relocated") from the first network site 200 to the control site 207 (*i.e.*, from the first domain 200 to the control site domain 207). This may be performed by transmitting a first message from the first network site 200 to the client 206 having a relocate command, a "find.sub.-- user" command that instructs the control site 207 to find information relating to the client 206, transient verification identifiers (passwords) negotiated between the first server 212 and the control server 214, and information indicating that the commands were issued by the first network site 200.

Receipt of the first message by the client 206 first causes the client 206 to relocate to the control site 207 (*i.e.*, the control domain) and then, when in the control site domain 207, to direct the find.sub.-- user command and first network site information to the control site 207. Upon receipt of the find.sub.-- user command and first network site information, the control site 207 responsively executes a plurality of steps (on behalf of the first network site 200) that further implement preferred embodiments of the invention. One of those steps causes the control site 207 to responsively interact with the client 206, in a conventional manner, to determine if the browser 210 includes a control site cookie (*i.e.*, control site data block) from the control site 207 (step 308).

The cited text of *Buckland* does not teach or suggest a distribution server.

Similarly, the Examiner has also cited *Brendel* as teaching, at Figure 8 and Column 10, lines 38-53, Applicant's claimed feature of "responsive to determining that the client's request to receive the file from the content server did not originate as the reference from one of the set consisting of the load distribution server and the content server, sending to the client a file requesting that the client contact the load distribution server". The cited text of *Brendel* discloses:

FIG. 8 is a diagram illustrating TCP state migration of a connection from the load balancer to a server node. Browser 10 connects through Internet 66 to load balancer 70 and sends a URL request 102 once

the connection 100 is made. Load balancer 70 does not have to be a separate, dedicated router or PC, and is shown as a software application running on server 56. Load balancer 70 can use many variations of balancing algorithms to determine which server 56, 51, 52 should service the new URL request 102. Load balancer 70 determines that the request should be assigned to server 52. The connection and URL request are migrated from load balancer 70 to server 52 using TCP state migration 120. Server 52 accesses disk 62 to read requested file 26 and sends a copy of requested file 26 to browser 10 through Internet 66 as data transfer 104.

The Examiner asserts that “Brendel, in related prior art, teaches a load balancer (70) on the server 56 (figure 8, col.10, lines 38-53) wherein the load balancer receives, keeps track, assigns and delivers all incoming requests from client to the assigned servers.” Upon review of the cited text of *Brendel* and the reasoning of the Examiner, and upon review of the relevant claim, Applicant respectfully submits that the reference does not disclose “responsive to determining that the client's request to receive the file from the content server did not originate as the reference from one of the set consisting of the load distribution server and the content server, sending to the client a file requesting that the client contact the load distribution server.” First, the cited text does not disclose “sending to the client a file requesting that the client contact the load distribution server”, because the cited text of *Brendel* discloses a client initially and directly contacting a load balancer. Similarly, and also for the same reason, Applicant respectfully submits that the cited text does not teach or disclose performing the sending step “responsive to determining that the client's request to receive the file from the content server did not originate as the reference from one of the set consisting of the load distribution server and the content server”. In each case, Applicant respectfully submits that the cited text, which discusses a client directly (and without prompting) contacting a load balancer, never suggests instructing a client to contact a load balancer.

Applicant respectfully observes that neither of *Brendel* and *Buckland*, nor the combination of *Brendel* and *Buckland* teaches or suggests Applicant's recited feature of “responsive to determining that the client's request to receive the file from the content server did not originate as the reference from one of the set consisting of the load distribution server and the content server, sending to the client a file requesting that the client contact the load distribution server.”

Further, Applicant respectfully submits that the combination of the references does not suggest to one skilled in the art to modify the teachings of the references to obtain Applicant's claimed invention. Even if the references could be combined, as is suggested by the Examiner, Applicant respectfully submits that the combination of the cited texts merely discloses planting and seeking a cookie when a client contacts a server or a load balancer, which does not suggest "responsive to determining that the client's request to receive the file from the content server did not originate as the reference from one of the set consisting of the load distribution server and the content server, sending to the client a file requesting that the client contact the load distribution server." In view of the absence, when considering a possible combination of all of the cited references, of the recited functionality, Applicant respectfully submits that the rejection under 35 U.S.C. § 103 is overcome.

V. Arguments with respect to Claim 1 apply broadly

Applicant respectfully submits that the rejection of exemplary Claim 1 under 35 U.S.C. § 103 is overcome. The foregoing arguments made with respect to Claim 1 are also made with respect to Claims 2-8, which further limit and patentably distinguish Claim 1. The foregoing arguments are also made with respect to Claims 9 and 17, which claim a computer program product and a system for performing Applicant's invention, respectively. The foregoing arguments are similarly made with respect to Claims 10-16, which further limit and patentably distinguish Claim 9 and with respect to Claims 18-24, which further limit and patentably distinguish Claim 17.

VI. The "updating" step of Applicant's Claim 4 is not taught or suggested

With respect to exemplary Claim 4, Applicant respectfully submits that the cited combination of references does not teach or suggest Applicant's claimed feature of "offering in the file requesting that the client contact the load distribution server a means to update a bookmark file to include the load distribution server", as recited in Applicant's exemplary Claim 4. The Examiner correctly notes that "**Buckland and Brendel** does not explicitly teach a means to update a bookmark file to include the load distribution server", and then cites *Nielsen*, at Column 9, lines 14-60 and Column 12, line 15-column 13, line 61, as teaching the recited functionality. The cited text of Column 9 of *Nielsen* discloses:

FIG. 3 illustrates a Web Page provided by a WWW Server apparatus **202** executing a WWW Server application **217** as viewed on a display device **147** using a WWW Browser application **215**. This particular WWW Browser application **215** is provided by Netscape Communications, Inc. This Netscape browser application is one of many possible WWW Browser applications that would be improved by incorporating the invention therein. The Web Page information **301** is displayed in a window **303** by the WWW Browser application **215**. This WWW Browser application **215** provides operator command buttons **307**, WWW navigation buttons **309**, and presents the URL for the currently displayed Web Page **311**. This Figure shows the bookmark popup **313**. This popup **313** displays the titles of current bookmarked Web Pages **315**, **317**, **319** along with a menu command **321** used to bookmark the current page **301**. The bookmark labeled as **319** is the prior art method of displaying the bookmark. The bookmark labeled as **315** indicates, by the ".cndot." prefix, that the bookmark has changed as discovered by the "What's New" command. The prior art does not provide this capability. Finally, the bookmark labeled as **317** indicates, by enclosing the Web Page title within ">>" and "<<" that the Web Page has sufficiently changed and that the Web Page has not be subsequently viewed. A preferred embodiment of the invention also provides the capability of supporting a floating window **323** used to list only the titles of the Web Pages **325** that have received notification of sufficient change but that have not been subsequently viewed.

Most WWW Browser applications also provide a bookmark management facility. The Netscape browser provides a bookmark window that allows a user to organize the bookmarks. **FIG. 4** illustrates a typical bookmark window **401** listing a number of bookmarked Web Page titles **403**, each with an associated status icon **405**. Status icons like the one labeled as **411** indicate that the Web Page has either not been modified or that the "What's New" process has not been applied to the bookmark. Web Pages that have changed and detected by the "What's New" process are marked by an icon such as the one labeled as **407**. Web Pages that have been sufficiently changed and for which the WWW Browser application has automatically received notification of the change but that have not been viewed since the notification are marked with an icon such as the one labeled as **409**. One skilled in the art will understand that there are many equivalent ways to organize this window **401** and many equivalent ways to indicate the status of each bookmark.

The cited text of column 12 discloses:

FIGS. 9A and 9B illustrate the process used by the WWW Server application to notify subscribers of a sufficient change to a Web Page. The process starts at the terminal labeled as **901**. Once the maintainer of the Web Page has modified **903** the Web Page and submits it to service, the process checks whether the Web Page has any notification subscribers **905**. This is accomplished by searching the Server's Notification Subscriber database for a record **500** having a "URL" field **501** matching the URL of the modified page. If no matching record is found **905**, the process completes through the terminal labeled as **907**. If **905** a match is found, the process then **911** determines whether the maintainer of the Web Page considers the modification to be sufficient. This determination can be accomplished by displaying a dialog to the user, by use of a command line switch or any of a number of equivalent methods well known in the art. If **911** the maintainer does not consider the changes to the Web Page to be sufficient, the process completes through the terminal labeled as **907**.

If the maintainer considers the changes to the Web Page to be sufficient **911** (thus initiating the sending of notification messages to all notification subscribers), the process continues through the terminal labeled as **915** to the terminal labeled as **917** of **FIG. 9b**. Each record **500** of the Server's Notification Subscriber database is examined **919**. When all the records **500** in the Server's Notification Subscriber database have been examined, the process completes through the terminal labeled as **921**.

Each **919** record **500** in the database is first checked **923** whether the URL of the modified Web Page matches the URL stored in the "URL" field **501** of the record **500**. If the URLs do not match, the next record **500** is examined as indicated by the arrow labeled as **924**.

If **923** the URLs match, an update notification e-mail message is constructed and sent **925** to the intended recipient's address that is contained in the "User's E-mail Address" field **503** of the record **500**. This e-mail message has the form shown in Table 1.

TABLE 1

To: <user's e-mail address>
From: <email address for the notification agent at the server>
Date: <current date and time>
Subject: Web page updated: <URL>
X-Web-Update-Notification: <URL>
The Web Page entitled <pagetitle>
at <URL> has been updated.
To stop getting these update notifications, please simply reply
to this message with the single word
Stop
in the body of the message.

Where "<URL>" is replaced by the actual URL string of the modified Web Page; "<pagetitle>" is replaced by the title of the Web Page as included in the HTML definition of the Web Page; and the "<user's e-mail address>" is replaced by the intended recipient's e-mail address.

Next, if **927** an e-mail message from the WWW Server application addressed to this intended recipient about this URL has previously been sent during the current day, the process continues **919** with the next record **500** as indicated by the arrow labeled as **928**. To determine whether e-mail has been sent to this intended recipient during the current day, the Server scans records **510** in the Server's Contact database for any record **510** matching the recipient's e-mail address in the "User's E-mail Address" field **511**, the URL in the "URL" field **509** and containing the current date in the "Date Update Message First Sent" field **515**.

If **927** the WWW Server application has not already sent e-mail to this recipient on this current date, a record **510** is constructed and added **929** to the Server's Contact database. The "URL" field **509** is initialized to the "URL" field **501** of the current record **500** from the Server's Notification Subscriber database. The "User's E-mail Address" field **511** of this record **510** is initialized to the string of the "User's E-mail Address" field **503** of the current record **500** from the Server's Notification Subscriber database. The "Error From User" field **513** of the record **510** is initialized to FALSE and the "Date Update Message First Sent" field **515** is set to the current date. Finally, the process continues **919** to the next record **500** as indicated by the arrow labeled as **930**.

User's E-mail Processing

FIG. 10 illustrates the method used by the e-mail facility in the recipient's computer to process an update notification message sent from the WWW Server application. The process starts at the terminal labeled as **1001**. The e-mail message is received **1003** by the recipient's e-mail facility **201**. The recipient's e-mail facility **201** examines the message to determine whether **1005** the message is a Web Update Notification Message as illustrated in Table 1. This determination is performed by examining the header portion of the message for the X-Web-Update-Notification: header. If this header is not contained in the header portion of the message, the e-mail system processes the message in the normal manner as indicated by the terminal labeled as **1007**. However **1005**, if the message is an update notification message, the process sends **1009** the relevant information to the WWW Browser application as indicated by the dashed arrow labeled as **1014**. This communication uses an interprocess communication mechanism such as SUN's ToolTalk, Apple's AppleScript,

Microsoft's OLE, TCP/IP or some operating system supported interprocess communication facility. The WWW Browser application processes this information and flags the appropriate bookmark as sufficiently modified. Next 1011, the update notification message is removed from the user's e-mail system so that the user will not view the message. Finally, the process completes through the terminal labeled as 1013.

The data transferred between the e-mail facility 201 and the WWW Browser application 215 using the interprocess communication link 227 comprises: a value that identifies the data as a bookmark update notification; the URL from the field-body of the X-Web-Update-Notification: header; and the date and time value from the field-body of the Date: header.

Having examined the cited texts, Applicant respectfully submits that the Applicant's recited feature of "offering in the file requesting that the client contact the load distribution server a means to update a bookmark file to include the load distribution server" is not taught or suggested. Applicant has likewise been unable to identify analogous concepts in the cited texts. The Examiner alleges:

"Nielsen, in the related art, teaches the feature of sending an email notification to inform client updating a bookmark file when there is a sufficient changes to a web page (figure 3, col. 9, lines 14-60, col. 12, line 15-col. 13, line 61).

It would have been obvious to one skill in the art at the time of the invention was made to incorporate the feature of updating a bookmark file, as disclosed by Nielsen, into the system of Buckland and Brendel to include a means to update bookmark file because it were conventionally employed in the art to provide a useful and enhance system that monitor the sufficient changes of bookmarked information file/Web page so that the user can be notified and update bookmark of the changed information file/Web page (see Nielsen col 1., lines 6-14, col. col. 4, lines 12-39)"

Applicant respectfully submits that the Examiner has not shown any reference that teaches or suggests "offering in the file requesting that the client contact the load distribution server a means to update a bookmark file to include the load distribution server". Applicant respectfully traverses the Examiner's assertion that "It would have been obvious to one skill in the art at the time of the invention was made to incorporate the feature of updating a bookmark file, as disclosed by Nielsen, into the system of Buckland and Brendel to include a means to

update bookmark file because it were conventionally employed in the art". *Nielsen* does not disclose "offering in the file requesting that the client contact the load distribution server a means to update a bookmark file to include the load distribution server". If anything, the combination proposed by the Examiner would result in a technological chimera that emails the user of a device when the content of a web page on a load distribution server changes. Applicant respectfully submits that the combination of references does not teach or suggest functionality addressed or suggested by none of them individually. Applicant respectfully submits that the rejection of Claims 4-6, 12-14, and 20-22 under 35 U.S.C. § 103 is overcome.

VII. Conclusion

It is respectfully submitted that the claims are in condition for allowance and favorable action is requested. A two month extension of time is believed to be required, and a check for the required fee is enclosed herewith. However, in the event that a further extension of time is required, please charge that extension fee and any other required fees to **IBM Corporation's Deposit Account Number 09-0447**.

Applicant respectfully requests the Examiner contact the undersigned attorney of record at (512) 542-3678 if such would further or expedite the prosecution of the present Application.

Respectfully submitted,



James E. Boice

Reg. No. 44,545

DILLON & YUDELL, LLP

8911 N. Capital of Texas Highway

Suite 2110

Austin, Texas 78759

(512) 343-6116

ATTORNEY FOR APPELLANT